We Claim:

1. A method for compacting state data of an emulation system, comprising steps of:
receiving a first sample of state data;
sorting the first sample;
determining if residual storage space in a first buffer exists; and
storing the sorted first sample.

2. The method of claim 1, further comprising a step of determining whether the first buffer is full after storing the sorted first sample in the first buffer.

3. The method of claim 1, further comprising a step of portioning the sorted first sample into two portions.

4. The method of claim 3, wherein the step of storing the sorted first sample comprises storing a portion of the sorted first sample in the first buffer, and storing a remaining portion of the sorted first sample in a second buffer.

5. The method of claim 4, wherein the portion of the sorted first sample fills the first buffer.

6. The method of claim 5, wherein the second buffer assumes the role of the first buffer and the first buffer assumes the role of the second buffer.

7. The method of claim 5, further comprising the step of draining the first buffer into an output storage device.

8. The method of claim 1, further comprising steps of:
receiving a second sample of state data;
sorting the second sample; and
storing the sorted second sample.

9. The method of claim 8, wherein the step of storing the sorted second sample comprises:

determining if residual storage space in the first buffer exists after storing the sorted first sample; and

storing at least a portion of the sorted second sample in the first buffer if residual storage space in the first buffer exists after storing the sorted first sample.

10. The method of claim 1, wherein the first sample of state data is received from a reconfigurable emulation resource.

11. The method of claim 1, wherein the step of storing the sorted first sample comprises storing the entire first sample in the current buffer.

12. The method of claim 1, wherein the sorted first sample comprises both data of interest and ignored data.

13. The method of claim 1, wherein the step of sorting the first sample comprises steps of:

identifying a bit of data of interest within the first sample;

identifying a bit position within the first buffer where residual storage space exists; and

sorting the bit of the identified data of interest of the first sample into a bit position of a sorted first sample,

wherein the identified bit position within the first buffer is the bit position of the sorted first sample.

14. The method of claim 13, further comprising steps of:

identifying a next bit position within the first buffer where residual storage space exists; and

sorting at least a second bit of the identified data of interest of the first sample into a next bit position of the sorted first sample,

wherein the next identified bit position within the first buffer is the next bit position of the sorted first sample.

15. The method of claim 13, wherein upon determining that residual storage space in the first buffer does not exist, the method further comprising a step of:

identifying a first bit position of a second buffer where residual storage space exists,

wherein the identified first bit position of the second buffer where residual storage space exists is a first bit position of the sorted first sample.

16. The method of claim 1, further comprising a step of storing information associated with the first sample.

17. The method of claim 16, wherein the information comprises a bit position of data of interest of the first sample.

18. The method of claim 16, wherein the information comprises an identification of a pin associated with the first sample.

19. The method of claim 16, further comprising steps of:

receiving a second sample of state data;

storing the second sample;

storing information associated with the first sample in memory; and

storing information associated with the second sample in memory.

20. An apparatus, comprising:

a first select logic device configured to receive samples of state data, to sort samples of state data, and to select data of interest from each of the samples of state data;

first and second buffers coupled to the first select logic device and configured to receive the selected data of interest;

a second select logic device coupled to the first and second buffers and configured to select the first and second buffers in an alternating manner to drain the selected buffer; and

an output storage device coupled to the second select logic device and configured to receive data drained from the selected buffer.

21. The apparatus of claim 20, wherein the first select logic comprises a multiplexer.

22. The apparatus of claim 20, wherein the second select logic device comprises a multiplexer.

23. The apparatus of claim 20, wherein the first select logic device sends the data of interest to the second buffer when the first buffer becomes full.

24. The apparatus of claim 20, wherein the first select logic device comprises a data of interest sorter.

25. The system of claim 20, further comprising:

a memory configured to store information associated with the samples of state data.

26. The system of claim 25, wherein the information comprises at least a bit position of data of interest of the sample of state data.

27. The system of claim 25, wherein the information comprises an identification of a pin associated with each of the samples of state data.

28. The system of claim 20, wherein the output storage device is configured to store information associated with each of the samples of state data.

29. A method for associating trace data chains with pins of an integrated circuit, the method comprising steps of:

determining a trace data fill rate of each of a plurality of trace data chains; and

determining a schedule for associating a plurality of pins with the plurality of trace data chains to transfer data out of the trace data chains based at least upon the determined trace data chain fill rates.

30. In an emulation debugging resource, a method comprising steps of:

determining fill rates of a plurality of trace data chains;

determining a schedule for associating a plurality of pins with the plurality of trace data chains based at least upon the determined fill rates; and

associating a set of the plurality of trace data chains with the plurality of pins in accordance with the determined schedule.

31. In an integrated circuit, the integrated circuit including an emulator, an apparatus comprising:

a plurality of trace data chains;

a trace pin select logic device coupled to the plurality of trace data chains to select a set of the plurality of trace data chains;

a plurality of pins; and

a memory coupled to the trace pin select logic device and configured to store a schedule to associate the selected set with the pins.